

Remarks

Entry of the amendments, reconsideration of this application in light of the remarks below and allowance of all pending claims are respectfully requested. Claims 65 - 78 remain pending in the application. It should be noted that applicant is not acquiescing to the stated rejections and that the addition of the new claims has been made solely in an effort to move the prosecution of this application forward to issue.

Claim 52, the only formerly pending claim which incorporated the phrase “said program being absent embedded debug commands” has been cancelled. No new claim submitted herewith includes this phrase. Accordingly, any claim rejection based on inclusion of the above-identified phrase has been negated. The Examiner is thanked for the explanation of his position with respect to use of the “said program being absent embedded debug commands” phrase.

Applicants have submitted a new set of claims herewith. Basically, the new independent claims 65, 71 and 75 correspond to former independent claims 45, 53 and 59 with each new independent claim incorporating the two dependent claims that were previously linked to them. For example, new independent claim 65 represents the combination of previously pending claims 45, 46 and 47.

Applicants also wish to point out that the independent claims have been modified to recite the use of specific attributes of machine instructions rather than just a general reference to attributes. Applicants respectfully submit that this change clarifies that any particular attribute of a machine instruction can be used for purposes of constructing applicants instruction profile in contrast to inclusion of a machine instruction which has attributes, which attributes are, however, not used to restore breakpoints in accordance with applicant’s invention.

The Office Action also noted that the claims do not positively recite that the breakpoint, once automatically restored, is inserted into the source code. Applicants, however, respectfully submit that the phrase “automatically restoring the breakpoint to the selected step of a modified program,” from new claim 65, clearly indicates that the

breakpoint is indeed restored to the selected step of the modified program meaning that the breakpoint is actually inserted into the modified version of the source code.

It is respectfully submitted that the newly submitted claims overcome the prior 35 U.S.C 112, first paragraph, based rejections for the foregoing reasons.

The former claims were rejected under 35 U.S.C. 103(a) as being unpatentable over U.S. Patent No. 6,263,489 to Olsen et al (hereinafter “Olsen”) in view of U.S. Patent 6,282,698 to Baker et al (hereinafter “Baker”).

Applicants would again reiterate and incorporate herein their prior characterization of Olsen and Baker as described in their prior Responses. Applicants respectfully submit that the Office Action description of the teachings of these two references is too expansive and unwarranted based on a straightforward reading of their content.

For example, Olsen creates a table wherein each machine code instruction is associated with a source code construct for which it was generated. That table does not rely on specific attributes of the associated machine code instruction. Applicant respectfully and further submits that this teaching in Olsen is directed to insertion of a breakpoint instruction into binary, not source code, is not automatic and does not use any characteristics of the replaced machine code instruction to determine the selected step. In fact, the table in Olsen that is relied on as a basis for asserting that Olsen creates an instruction profile is nothing more than a means of associating the inserted breakpoint instruction with the entire machine code instruction it replaced rather than any specific attributes of that instruction. This is merely a swap of information on an “as is” basis and not a profile determined by machine instruction attributes as is expressly specified in all of applicant’s claims.

In addition, it is important to note that in Olsen the restored breakpoint is inserted back into the same modified program at the location where it was originally inserted. This is clearly not the case with applicant’s invention where the instruction profile is based on a code instruction or more, associated with the first version of the source code. The breakpoint is restored into a source version of the modified or debugged program. In this respect, it should be noted no such breakpoint reinsertion takes place in Baker at any time.

The Office Action again argues that one of ordinary skill in the art would have been motivated to apply Olsen's teachings regardless of how or why a program is modified, to automatically restore a breakpoint to a selected step of a modified program, regardless of whether it is an optimization (as in Olsen), a modification of the source code (as in Baker), or some other operation that provides the modified program. That argument is apparently necessary because the hypothetical combination urged as obviating applicant's claimed invention lacks this necessary step. The motivation to automatically restore a breakpoint using applicant's instruction profile seems to have come from applicant, not the prior art. This point was previously discussed

With respect to the role played by one of ordinary skill in the art, the Examiner's attention is respectfully directed to the holding in *Ex parte Levengood*, 28 USPQ 2d 1300 (Bd. Pat. App. & Inter. 1993). *Levengood* held that the fact one of ordinary skill in the art has the capabilities to arrive at the invention is not the test for whether one of ordinary skill in the art would have arrived at the invention based on the teachings of the prior art; "That which is within the capability of one skilled in the art, is not synonymous with obviousness." Applicant respectfully submits that *Levengood* applies in all respects to the asserted basis of the current rejection.

According to §706.02(j) of the M.P.E.P., a rejection under 35 U.S.C. 103 requires several supporting criteria including an explanation of why one of ordinary skill in the art at the time the invention was made would have been motivated to make the proposed modification. While the Office Action included a conclusion that one of ordinary skill would have been motivated to modify and combine the prior art references, no explanation of the motivating factors was set forth. Applicants respectfully submit that the current rejection under 35 U.S.C. 103, therefore, requires further explanation in order to be effective.

§706.02(j) also states that a *prima facie* case of obviousness cannot be established unless three basic criteria are met. One - there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine the reference teachings and, other than a conclusionary statement about obviousness, no such suggestion or motivation has been

expressly made. Two - there must be a reasonable expectation of success, an outcome not apparent where the combination of Olsen and Baker, especially Baker with its reliance on brute force, slavish comparisons of binary code without requiring access to or use of source code in stark contrast to applicant's invention. Three – the prior art references must teach or suggest all of the claim limitations, which is not the case with respect to applicant's invention, an important distinction, since the teaching or suggestion to make the claimed combination must come from the prior art and not be a hindsight conclusion based on applicant's invention.

As for Baker, it is directed at determining how many instances of certain code attributes there are on a total basis in each of two program versions being compared and in then making a determination of similarity based on the gross results of such comparison. There are no breakpoints to be considered, tracked or reinserted and no source code is even used or needed, per Baker's Abstract. Even a person of ordinary skill in this art, having Olsen in hand, would not be inclined to seek out or utilize Baker since it offers no teaching or assistance in adding inventive elements missing from Olsen to create applicant's invention as claimed.

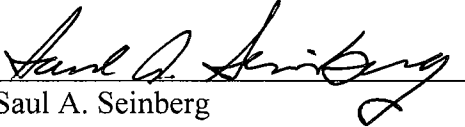
Applicant contends that the hypothetical combination of Olsen and Baker stops well short of the combination of elements claimed by applicant. As previously noted, Olsen and Baker are each trying to solve significantly different problems than applicants and are not reasonably pertinent to the particular problem that applicant's invention solves or to the express manner in which the invention is claimed.

Applicant wishes to respectfully point out that there has been no suggestion, motivation or teaching for those skilled in the art to combine Olsen and Baker in the manner used in the Office Action. Olsen is directed at a situation involving optimized code while Baker is concentrated exclusively on comparing large sets of code, if not entire programs, to see if they are different. Accurately relocating breakpoints and the absence in either reference of an instruction profile used for this purpose as expressly called for in applicant's pending claims patentably separates applicant's invention from the hypothetical combination proposed in the Office Action.

For all of the above reasons, and in light of the newly added claims now pending in this case, applicant respectfully submits that all such claims are patentable over the combination of Olsen and Baker and are of at least the required distinctiveness.

Should the examiner wish to discuss this case with applicant's attorney, please contact applicant's attorney at the below listed number.

Respectfully submitted,

A handwritten signature in dark ink, appearing to read "Saul A. Seiberg", is written over a horizontal line.

Saul A. Seiberg
Attorney for Applicant
Registration No.: 24,840

Dated: March 19, 2007.

HESLIN ROTHENBERG FARLEY & MESITI P.C.
5 Columbia Circle
Albany, New York 12203-5160
Telephone: (518) 452-5600
Facsimile: (518) 452-5579